

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2000-244358

(P2000-244358A)

(43) 公開日 平成12年9月8日 (2000.9.8)

(51) Int.Cl. ⁷	識別記号	F I	テーマコード [*] (参考)
H 0 4 B 1/38		H 0 4 B 1/38	
H 0 4 Q 7/38		H 0 3 M 1/12	C
// H 0 3 M 1/12		H 0 4 B 7/26	1 0 9 H

審査請求 未請求 請求項の数14 O L (全 15 頁)

(21) 出願番号	特願平11-351926	(71) 出願人	000003078 株式会社東芝 神奈川県川崎市幸区堀川町72番地
(22) 出願日	平成11年12月10日 (1999. 12. 10)	(72) 発明者	吉田 弘 神奈川県川崎市幸区小向東芝町1番地 株 式会社東芝研究開発センター内
(31) 優先権主張番号	特願平10-362059	(72) 発明者	菅野 伸一 神奈川県川崎市幸区小向東芝町1番地 株 式会社東芝研究開発センター内
(32) 優先日	平成10年12月21日 (1998. 12. 21)	(74) 代理人	100076233 弁理士 伊藤 進
(33) 優先権主張国	日本 (J P)		

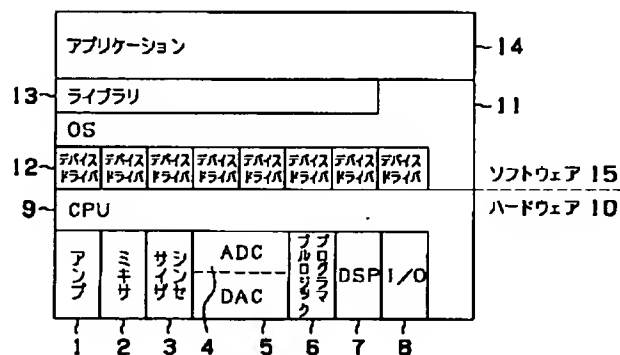
最終頁に続く

(54) 【発明の名称】 無線通信装置、無線通信方法及び記録媒体

(57) 【要約】

【課題】 汎用性を有し、ハードウェアアーキテクチャに依存しないアプリケーションプログラムを利用可能な無線通信装置、その無線通信方法及びそのプログラムが記録された記録媒体を提供する。

【解決手段】 受信した無線信号をデジタル信号に変換し、送信するデジタル信号を無線信号に変換し、AD変換手段の出力に対するデジタル信号処理と、DA変換手段の入力に対するデジタル信号処理とを行う、デジタル信号処理手段と、所定の無線機能を実現するためのソフトウェア群からなるライブラリを参照して、所定のソフトウェアを基本ソフトウェア上で実行することにより、少なくとも無線信号の送受信、AD変換、DA変換、デジタル信号処理のいずれかを制御する。



1

【特許請求の範囲】

【請求項 1】無線信号を送受信する送受信手段と、
受信した前記無線信号をデジタル信号に変換して出力する AD 変換手段と、
送信するデジタル信号を前記無線信号に変換して出力する DA 変換手段と、
前記 AD 変換手段の出力に対するデジタル信号処理及び前記 DA 変換手段の入力に対するデジタル信号処理を行うデジタル信号処理手段と、

所定の無線機能を実現するためのソフトウェア群からなるライブラリを参照して、所定のソフトウェアを基本ソフトウェア上で実行することにより、前記送受信手段と、前記 AD 変換手段と、前記 DA 変換手段と、前記デジタル信号処理手段との少なくとも 1 つを制御する制御手段とを具備したことを特徴とする無線通信装置。

【請求項 2】前記送受信手段は、アナログ信号処理を行うアナログデバイスを具備し、
前記ライブラリは、少なくとも前記アナログデバイスを制御するためのソフトウェア群を有することを特徴とする請求項 1 に記載の無線通信装置。

【請求項 3】前記制御手段は、
前記基本ソフトウェアを介して、前記所定のソフトウェアを動作させることで発生したデータを、前記送受信手段を駆動するデバイスドライバへ転送する機能を更に具備したことを特徴とする請求項 1 に記載の無線通信装置。

【請求項 4】無線信号を送受信する送受信手段と、
受信した前記無線信号をデジタル信号に変換して出力する AD 変換手段と、
送信するデジタル信号を前記無線信号に変換して出力する DA 変換手段と、

前記 AD 変換手段の出力に対するデジタル信号処理と、前記 DA 変換手段の入力に対するデジタル信号処理とを行う、デジタル信号処理手段と、
所定の無線機能を実現するためのソフトウェア群からなる第 1 のライブラリを記憶する第 1 の記憶手段と、
前記第 1 のライブラリと本無線通信装置以外にあって前記第 1 のライブラリとは異なる第 2 のライブラリとを参照して、所定のソフトウェアを基本ソフトウェア上で実行することにより、前記送受信手段と、前記 AD 変換手段と、前記 DA 変換手段と、前記デジタル信号処理手段との少なくとも 1 つを制御する制御手段とを具備したことを特徴とする無線通信装置。

【請求項 5】所定のソースファイル及びコンパイラを記憶する第 2 の記憶手段を更に具備し、
前記制御手段は、前記コンパイラを用いて前記所定のソースファイルをコンパイルするときに前記第 1 及び第 2 のライブラリをリンクさせて前記所定のソフトウェアを得ることを特徴とする請求項 4 に記載の無線通信装置。

【請求項 6】前記制御手段は、所定のコンパイラによつ

2

て前記第 2 のライブラリを参照しながら所定のソースファイルをコンパイルして得た実行ファイルをダウンロードし、ダウンロードした実行ファイルと前記第 1 のライブラリとのリンクによって前記所定のソフトウェアを得ることを特徴とする請求項 4 に記載の無線通信装置。

【請求項 7】無線信号を送受信する第 1 の処理と、
受信した前記無線信号をデジタル信号に変換して出力する第 2 の処理と、

送信するデジタル信号を前記無線信号に変換して出力する第 3 の処理と、

前記第 2 の処理の出力に対するデジタル信号処理と、
前記第 3 の処理の入力に対するデジタル信号処理とを行う第 4 の処理と、

所定の無線機能を実現するためのソフトウェア群からなるライブラリを参照して、所定のソフトウェアを基本ソフトウェア上で実行することにより、前記第 1 の処理と、前記第 2 の処理と、前記第 3 の処理と、前記第 4 の処理との少なくとも 1 つを制御する第 5 の処理とを具備したことを特徴とする無線通信方法。

【請求項 8】コンピュータが読み出し可能なプログラムを格納した記録媒体において、
前記プログラムは、

所定の無線機能を実現するためのソフトウェア群からなるライブラリを参照して、所定のソフトウェアを実行する第 1 の処理と、

前記第 1 の処理で実行した前記所定のソフトウェアにより、無線信号を送受信する機能と、受信した前記無線信号をデジタル信号に変換して出力する機能と、送信するデジタル信号を前記無線信号に変換して出力する機能と、前記デジタル信号に対するデジタル信号処理を行う機能とのいずれかを制御する第 2 の処理とを具備したことを特徴とする記録媒体。

【請求項 9】無線信号を送受信するための送受信処理を行うアナログハードウェアを制御するために前記アナログハードウェアに依存して記述されたプログラムであるライブラリを参照して、前記無線信号の送受信処理を実行させるためのプログラムであって、前記アナログハードウェアに依存することなく記述されたプログラムを記録したコンピュータ読み取り可能な記録媒体。

【請求項 10】前記送受信手段は、アナログ信号処理を行うアナログデバイスとデジタル信号処理を行うデジタルデバイスとを具備し、

前記制御手段は、同一機能が前記アナログデバイスとデジタルデバイスとの双方で実現可能な場合には、所定の条件に従って前記アナログデバイス又は前記デジタルデバイスの一方を選択して機能を達成するか又は、前記アナログデバイス又は前記デジタルデバイスの一方を優先的に選択して機能を達成することを特徴とする請求項 1 に記載の無線通信装置。

【請求項 11】前記制御手段による選択は、前記ライブ

3

ラリを参照することでソフトウェア的に実行されることを特徴とする請求項 10 に記載の無線通信装置。

【請求項 12】前記送受信手段は、アナログ信号処理を行うアナログデバイスとデジタル信号処理を行うデジタルデバイスとを具備し、

前記アナログデバイス及びデジタルデバイスのうちの少なくとも一方の手段は、仕様に関する情報を記憶する仕様記憶部を設け、

前記制御手段は、前記仕様に関する情報を読み取り、読み取った情報に基づいて前記制御を行うことを特徴とする請求項 1 に記載の無線通信装置。

【請求項 13】アナログデバイスを含み無線信号の送受信を行うための送受信手段を有するハードウェアと、前記ハードウェアに依存した基本ソフトウェア上で動作し前記無線信号の送受信を行うための命令群を備えたアプリケーションと、

前記ハードウェアに依存し、前記アプリケーションが備えた命令群を前記ハードウェアを制御する命令群に変換することによって前記アプリケーションが備える命令群が前記ハードウェアに依存しない場合でも、前記アプリケーションによる前記ハードウェアの制御を可能にするライブラリとを具備したことを特徴とする無線通信装置。

【請求項 14】アナログデバイスを含み無線信号の送受信を行う送受信手段と、

前記送受信手段をデジタル制御するプロセッサと、前記プロセッサ上で動作するオペレーションシステムと、

前記オペレーションシステム上で動作するアプリケーションと、

前記アプリケーションに基づく命令と前記送受信手段を制御する命令との間の変換を行うためのライブラリとを具備したことを特徴とする無線通信装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は無線通信装置に関し、特に無線機能を記述する専用言語を用いて無線機能をソフトウェア化する無線通信装置、無線通信方法及び記録媒体に関する。

【0002】

【従来の技術】現在携帯電話をはじめ移動通信システムに対する需要が非常に高くまた端末市場は活況を呈している。ところが、現在世の中にはいろいろな種類の通信システムが林立しているのが現状である。

【0003】例えば携帯電話では日本では PDC 方式、米国では AMPS 方式や CDMA 方式、欧州では GSM 方式と、さまざまな方式が混在している。さらに同じ日本国内においても携帯電話とは別に PHS というシステムが存在する。

【0004】IMT-2000 という世界統一方式の標

4

準化作業がすすめられているが、日米欧で全く同一のシステムとなる可能性は低い。

【0005】このような状況下において、複数のシステムに使用可能な、移動通信端末のマルチモード化に対する需要は非常に高い。しかしながら、端末におけるこのようなマルチモード化は非常に困難であり、物理的に 2 種類の端末をひとつの筐体の中へ内蔵するという単純な構成によってデュアルモード化するのが限度であった。

【0006】無線通信装置の従来例について以下、図面を参照して説明する。

【0007】図 16 は、無線通信装置の従来例の構成を示す。図中 701 は RF (高周波) の信号を増幅するアンプである。702 は周波数変換をおこなうミキサである。703 はミキサ 702 にたいして信号を供給するシンセサイザである。704 はアナログ信号をデジタル信号に変換する AD 変換器である。705 はデジタル信号をアナログ信号に変換する DA 変換器である。706 は専用 IC (ASIC) である。707 はデジタル信号処理プロセッサ (DSP) である。708 は専用 IC (ASIC) である。709 は中央演算処理装置 (CPU) である。710 は I/O インターフェイスである。711 はアンプ 701 ~ I/O インターフェイス 710 からなるハードウェア (の集合) である。712 は OS である。713 は OS 712 上で動作するアプリケーションプログラムである。714 は DSP 707 上で動作するアプリケーションプログラムである。715 は OS 712 ~ アプリケーションプログラム 714 からなるソフトウェア (の集合) である。

【0008】図 16 に示す従来例では、無線機はアンプ 701、ミキサ 702、シンセサイザ 703、AD 変換器 704 及び DA 変換器 705 等のアナログ回路と、専用 IC (ASIC) 706、708、DSP 707 及び CPU 709 等のデジタル回路と、I/O インターフェイス 710 等とからなるハードウェア 711 と、CPU 709 上で動く OS 712 およびその OS 上で動くアプリケーションプログラム 713 や、DSP 用のプログラム 714 等からなるソフトウェア 715 から構成されていた。

【0009】次に図 16 を参照し、無線端末の従来例の動作について説明する。受信時は、図示しないアンテナから入力された高周波信号を RF 部 (図中 701、702、703) で周波数の低いベースバンド信号に変換し、AD 変換器 704 で所望信号をデジタル信号に変換し、DSP 707 や CPU 709 を用いて送信されてきた信号を復調する動作をおこなう。

【0010】送信時は、送信するもとの信号に対して CPU 709 や DSP 707 で変調の操作をおこない、その信号を D/A 変換器 705 でアナログ信号に変換し、RF 部で高周波信号に変換してアンテナより送信する動作をおこなう。

5

【0011】上述のDSPやCPUが実行するプログラムはソフトウェア712、713、714として無線端末の内部のメモリ等に保存されている。

【0012】またアナログ部に対する制御は例えばASIC708を介してアプリケーションプログラム713からおこなわれる。図ではアナログ部の制御はASIC708からなされるが、実際にはCPU709やDSP707などから直接制御される場合もある。

【0013】ここで、このような無線端末の従来例には次のような問題があった。すなわち無線端末の従来例ではアプリケーションプログラム713、714がDSP707あるいはCPU709専用のプログラムであり、汎用性がなかった。したがってこれらのプログラムはある特定のハードウェアアーキテクチャ専用のものであり、他の異なるハードウェアアーキテクチャを持つ無線機でそのまま利用することが不可能であった。

【0014】またアナログ回路からなるRF部に対する制御はASICやゲートアレイなどの専用のLSIを用いてなされるため、他の異なるハードウェアで利用することが不可能であった。

【0015】

【発明が解決しようとする課題】このように、上述した従来の無線通信装置においては、アプリケーションプログラムは、汎用性がなく、その利用はハードウェアに依存した特定目的に限定されるという問題点があった。

【0016】本発明はかかる問題点に鑑みてなされたものであって、汎用性を有し、ハードウェアアーキテクチャに依存しないアプリケーションプログラムを利用可能な無線通信装置、その無線通信方法及びそのプログラムが記録された記録媒体を提供することを目的とする。

【0017】

【課題を解決するための手段】本発明の請求項1に係る無線通信装置は、無線信号を送受信する送受信手段と、受信した前記無線信号をデジタル信号に変換して出力するAD変換手段と、送信するデジタル信号を前記無線信号に変換して出力するDA変換手段と、前記AD変換手段の出力に対するデジタル信号処理及び前記DA変換手段の入力に対するデジタル信号処理を行うデジタル信号処理手段と、所定の無線機能を実現するためのソフトウェア群からなるライブラリを参照して、所定のソフトウェアを基本ソフトウェア上で実行することにより、前記送受信手段と、前記AD変換手段と、前記DA変換手段と、前記デジタル信号処理手段との少なくとも1つを制御する制御手段とを具備したものであり、本発明の請求項4に係る無線通信装置は、無線信号を送受信する送受信手段と、受信した前記無線信号をデジタル信号に変換して出力するAD変換手段と、送信するデジタル信号を前記無線信号に変換して出力するDA変換手段と、前記AD変換手段の出力に対するデジタル信号処理と、前記DA変換手段の入力に対するディジ

6

タル信号処理とを行う、デジタル信号処理手段と、所定の無線機能を実現するためのソフトウェア群からなる第1のライブラリを記憶する第1の記憶手段と、前記第1のライブラリと本無線通信装置以外にあって前記第1のライブラリとは異なる第2のライブラリとを参照して、所定のソフトウェアを基本ソフトウェア上で実行することにより、前記送受信手段と、前記AD変換手段と、前記DA変換手段と、前記デジタル信号処理手段との少なくとも1つを制御する制御手段とを具備したものであり、本発明の請求項7に係る無線通信方法は、無線信号を送受信する第1の処理と、受信した前記無線信号をデジタル信号に変換して出力する第2の処理と、送信するデジタル信号を前記無線信号に変換して出力する第3の処理と、前記第2の処理の出力に対するデジタル信号処理と、前記第3の処理の入力に対するデジタル信号処理とを行う第4の処理と、所定の無線機能を実現するためのソフトウェア群からなるライブラリを参照して、所定のソフトウェアを基本ソフトウェア上で実行することにより、前記第1の処理と、前記第2の処理と、前記第3の処理と、前記第4の処理との少なくとも1つを制御する第5の処理とを具備したものであり、本発明の請求項8に係る記録媒体は、コンピュータが読み出し可能なプログラムを格納した記録媒体において、前記プログラムは、所定の無線機能を実現するためのソフトウェア群からなるライブラリを参照して、所定のソフトウェアを実行する第1の処理と、前記第1の処理で実行した前記所定のソフトウェアにより、無線信号を送受信する機能と、受信した前記無線信号をデジタル信号に変換して出力する機能と、送信するデジタル信号を前記無線信号に変換して出力する機能と、前記デジタル信号に対するデジタル信号処理を行う機能とのいずれかを制御する第2の処理とを具備したものであり、本発明の請求項9に係る記録媒体は、無線信号を送受信するための送受信処理を行うアナログハードウェアを制御するために前記アナログハードウェアに依存して記述されたプログラムであるライブラリを参照して、前記無線信号の送受信処理を実行させるためのプログラムであって、前記アナログハードウェアに依存することなく記述されたプログラムを記録したものであり、本発明の請求項13に係る無線通信装置は、アナログデバイスを含み無線信号の送受信を行うための送受信手段を有するハードウェアと、前記ハードウェアに依存した基本ソフトウェア上で動作し前記無線信号の送受信を行うための命令群を備えたアプリケーションと、前記ハードウェアに依存し、前記アプリケーションが備えた命令群を前記ハードウェアを制御する命令群に変換することによって前記アプリケーションが備える命令群が前記ハードウェアに依存しない場合でも、前記アプリケーションによる前記ハードウェアの制御を可能にするライブラリとを具備したものであり、本発明の請求項14に係る無線通

7

信装置は、アナログデバイスを含み無線信号の送受信を行う送受信手段と、前記送受信手段をデジタル制御するプロセッサと、前記プロセッサ上で動作するオペレーションシステムと、前記オペレーションシステム上で動作するアプリケーションと、前記アプリケーションに基づく命令と前記送受信手段を制御する命令との間の変換を行うためのライブラリとを具備したものである。

【0018】本発明の請求項1において、送受信手段によって送受信する無線信号は、デジタル信号処理手段によってデジタル信号処理される。制御手段は、ライ
10 ブラリを参照して基本ソフトウェア上で所定のソフトウェアを実行することにより、送受信手段、AD変換手段、DA変換手段及びデジタル信号処理手段の少なくとも1つを制御する。

【0019】本発明の請求項4において、送受信手段によって送受信する無線信号は、デジタル信号処理手段によってデジタル信号処理される。制御手段は、第1
20 のライブラリだけでなく、本無線通信装置以外にあって第1のライブラリとは異なる第2のライブラリをも参照して基本ソフトウェア上で所定のソフトウェアを実行することにより、送受信手段、AD変換手段、DA変換手段及びデジタル信号処理手段の少なくとも1つを制御する。

【0020】本発明の請求項7において、第5の処理は、無線信号を送受信する第1の処理と、受信した無線信号をデジタル信号に変換して出力する第2の処理
と、送信するデジタル信号を無線信号に変換して出力する第3の処理と、第2の処理の出力に対するデジタル信号処理及び第3の処理の入力に対するデジタル信号処理を行う第4の処理との少なくとも1つを、所定の
30 無線機能を実現するためのソフトウェア群からなるライブラリを参照して、所定のソフトウェアを基本ソフトウェア上で実行することにより制御する。

【0021】本発明の請求項8において、記録媒体は、所定の無線機能を実現するためのソフトウェア群からなるライブラリを参照して、所定のソフトウェアを実行する第1の処理のためのプログラムと、第1の処理で実行した所定のソフトウェアにより、無線信号を送受信する機能と、受信した前記無線信号をデジタル信号に変換して出力する機能と、送信するデジタル信号を無線信
40 号に変換して出力する機能と、デジタル信号に対するデジタル信号処理を行う機能とのいずれかを制御する第2の処理のためのプログラムを有する。

【0022】本発明の請求項9において、記録媒体は、無線信号を送受信するための送受信処理を行うアナログハードウェアを制御するために前記アナログハードウェアに依存して記述されたプログラムであるライブラリを参照して、前記無線信号の送受信処理を実行させるためのプログラムが記録される。ライブラリはアナログハード
50 ウェアに依存して記述されているが、記録媒体に記録

8

されたプログラムはアナログハードウェアに依存することなく記述されており、アナログハードウェアの構成に拘わらず共通のプログラムが使用可能である。

【0023】本発明の請求項13において、ハードウェアはアナログデバイスを含んでおり、無線信号の送受信を行う。一方、アプリケーションは、ハードウェアに依存した基本ソフトウェア上で動作し無線信号の送受信を行うための命令群を備えている。アプリケーションはハードウェアに依存したライブラリを参照して、基本ソフトウェア上で実行される。ライブラリは、アプリケーションが備えた命令群をハードウェアを制御する命令群に変換するものであり、アプリケーションが備える命令群がハードウェアに依存しない場合でも、アプリケーションによるハードウェアの制御を可能にする。

【0024】本発明の請求項14において、送受信手段は、アナログデバイスを含んでおり、無線信号の送受信を行う。送受信手段は、プロセッサ上でオペレーションシステムを動作させ、オペレーションシステム上でアプリケーションを動作させることによって、プロセッサからデジタル制御される。この場合には、アプリケーションに基づく命令と送受信手段を制御する命令との間の変換がライブラリによって行われる。

【0025】

【発明の実施の形態】以下、図面を参照して本発明の実施の形態について詳細に説明する。図1は本発明の第1の実施の形態を説明するための説明図である。

【0026】図中1はRFの信号を増幅するアンプである。2は周波数変換をおこなうミキサである。3はミキサ2にたいして信号を供給するシンセサイザである。4はアナログ信号をデジタル信号に変換するAD変換器である。5はデジタル信号をアナログ信号に変換するDA変換器である。6はプログラマブルロジック（FPGA）である。7はデジタル信号処理プロセッサ（DSP）である。8はI/Oインターフェイスである。9は中央演算処理装置（CPU）である。10はアンプ1～CPU9からなるハードウェアである。

【0027】11はOSである。12はハードウェアの動作をOSから制御するためのデバイスドライバである。13はさまざまな関数群が記述されたライブラリである。14はアプリケーションである。15はOS11～アプリケーション14からなるソフトウェアである。

【0028】次に本実施の形態である無線端末の動作について説明する。

【0029】図1のようにCPU9上でOS11が動作し、このOS11が無線機全体の動作を制御する。OS11はCPU9に依存した基本ソフトウェアである。CPU9以外のハードウェア、例えばアンプ1、ミキサ2、シンセサイザ3、ADC4、DAC5、FPGA6、DSP7、I/Oインターフェイス8と、OS11の間にはデバイスドライバ12が介在し、このデバイス
50

ドライバ12は、OS11より出力された命令をそれぞれのハードウェアが実行可能な命令に変換しCPU9を経由してデータを送出し、また、各ハードウェアから状態を示す出力があればそれはCPU9を介してデバイスドライバ12に引き渡されてOS11が解釈できる信号に変換する役割を果たす。

【0030】したがって、デバイスドライバ12はOS11とハードウェア10に密接に関係があり、ハードウェア10が異なればデバイスドライバ12はそれに合わせて異なるものが必要となり、またOS11が変わればそれに対応したデバイスドライバ12が必要になる。

【0031】ライブラリ13はハードウェア10を制御し、あるいは機能そのものを実現するソフトウェアの集合体であり、複数の関数（命令セット）からなる。ライブラリ13はアプリケーション14から出される命令の中で汎用性が高い機能、およびハードウェア構成に依存する関数の集合である。したがってライブラリ13はハードウェア10やOS11に依存したものとなる。

【0032】以上述べたOS11、デバイスドライバ12およびライブラリ13は、ハードウェア10に依存するソフトウェアであり、これらをまとめてここでは基本ソフトウェアと呼ぶ。

【0033】アプリケーション14は無線機能を記述する専用言語（ここでは以下「radiodescription language」と呼ぶ）を用いて記述して得たプログラムであり、その記述はハードウェア10に依存しない。すなわち、アプリケーション14はハードウェア10が変わっても同じプログラムが利用できる。

【0034】異なるハードウェア10に対する制御の相違は、上述のデバイスドライバ12、OS11、ライブラリ13によって吸収することができ、アプリケーション14の変更は不要である。換言すると、アプリケーション14とOS11あるいはライブラリ13との間のインターフェイスが同じであればアプリケーション14は異なるハードウェア上で動作可能となる。アプリケーション14は上述したように無線記述言語を用いて記述されて得られる。

【0035】次に図2を用いて本実施の形態の無線機のソフトウェアの実行方法を説明する。

【0036】図2は、本実施の形態のソフトウェアの動作を説明する説明図である。図2において、図1と同等の構成を示すものについては同一の番号を付して説明する。図2中、10は無線機のハードウェアを示しており、図1におけるアンプ1～CPU9等から構成される。11はハードウェア10上で動作するOSである。20は無線機能が記述されたアプリケーションプログラムであり、図2の例ではソースファイルを示している。12はデバイスドライバであり、アプリケーションなどによってOS11に出された命令をハードウェア10に対する命令へ変換する動作をおこなう。13は関数など

が記述されたライブラリである。16はアプリケーションプログラム20を実行ファイル（Executable File）17に変換するコンパイラである。

【0037】なお、図1のアプリケーション14は、図2における実行ファイル17に相当する。しかし、アプリケーションプログラム20がコンパイル不要なインタプリタ言語等で記述されている場合には、アプリケーションプログラム20は図1のアプリケーション14そのものである。

10 【0038】無線記述言語で記述されたアプリケーションプログラム20は、実行する前にコンパイルされる。コンパイルはソースプログラムのOSが解釈できる機械語への変換を意味し、コンパイラ16を用いておこなうことによってなされる。

【0039】コンパイラ16によってコンパイルされた実行ファイル17はOS11上で実行される。実行時に実行ファイル17に記述された命令に応じてライブラリ13から必要な関数を引いて来て利用することによって実際の無線機能がソフトウェア的に実現される。

20 【0040】ライブラリ13には無線機能をハードウェア上で実現するための制御情報を生成するプログラムや、FPGAやDSPなどのプログラムそのものが予め関数として用意されており、アプリケーションによって与えられた関数とパラメータに基づいて、必要なプログラムを適切なパラメータとともにOS11へ引き渡す。

【0041】このライブラリ13は、あるハードウェア構成の場合はハードウェア的に実現されていた機能が、別のハードウェア構成の場合はソフトウェアで実現するような場合があるなど、ハードウェア構成によって異なるものとなる。

30 【0042】したがってライブラリ13はハードウェアに依存したものとなり、ハードウェア10が変更されるとそれに伴って変更される性質のものである。

【0043】また上述の方法を用いた場合、用意しておくライブラリが膨大なものとなる場合がある。しかしながらこのライブラリを全て無線機側で保持しておく必要はなく、頻繁に利用されるライブラリのみを無線機側で持ち、それ以外のライブラリは外部のものを利用し、コンパイル時にリンクしてしまう方法もある。

40 【0044】このようにライブラリを構成することで、無線機で持たなければならないライブラリを必要最小限に留めることが可能となる。

【0045】この場合の流れを図3に示す。図3のように、ここではライブラリは、無線機側にもつ第1のライブラリ31と外部にもつ第2のライブラリ32に分けられ、コンパイル時に第2のライブラリ32がリンクされ実行ファイルが作成され、この実行ファイルの実行時に第1のライブラリ31が参照されることによって無線機としての動作を実現する。

50 【0046】頻繁に参照される関数を第1のライブラリ

に用意し、残りの関数を第2のライブラリに用意することで、無線機側で持たなければならないライブラリの規模を大幅に縮小することができる。

【0047】ところで、図2及び図3の例では、アプリケーションプログラム20及びコンパイラ16を設ける位置は、限定されていない。図1のアプリケーション14に相当する実行ファイル17が無線機内のメモリに保持されるのに対し、アプリケーションプログラム20（ソースファイル）及びコンパイラ16は、無線機内に保持してもよく、無線機外部に保持してもよい。

【0048】図2の例では、アプリケーションプログラム20に対してコンパイラ16を用いたコンパイル処理によって作成した実行ファイル17をメモリに格納し、このメモリを無線機の製造時に無線機に搭載することが一般的である。

【0049】これに対し、図3の例では、第2のライブラリ32を機能拡張用と考え、その性質上、第2のライブラリ32のリンク処理及びコンパイル処理は、無線機の販売後に実施されることになる。無線機のメモリ容量、CPUの負荷等を考慮すると、アプリケーションプログラム20及びコンパイラ16を第2のライブラリ32と共に、無線機外部に用意した方がよい。例えば、アプリケーションプログラム20及びコンパイラ16を基地局等に用意し、基地局等においてアプリケーションプログラム20をコンパイルすると共に第2のライブラリ32をリンクし、これによって生成した実行ファイル17を無線機にダウンロードさせる。

【0050】なお、アプリケーションプログラム20及びコンパイラ16を無線機内のメモリに保持し、基地局等からダウンロードした第2のライブラリを用いて、無線機内で実行ファイル17を生成するようにしてもよいことは明かである。

【0051】次にOS11の動作を説明する。OS11はコンパイルされた実行ファイル17とライブラリ13によって与えられる命令をハードウェアブロックのデバイスドライバ12へ振り分ける作業をおこなう。

【0052】実行ファイルから出力された命令が、DSPに対するものであればDSP7のデバイスドライバに動作プログラムを送り、またミキサへの制御命令であればミキサ2のデバイスドライバに制御情報を送出する。

【0053】デバイスドライバ12はOS11を介して実行ファイル17（およびライブラリ13）によって与えられた命令をハードウェアレベルでの命令に変換する役割を果たす。この命令は最終的にはCPU9を介して各ハードウェアに対して信号を送り、所望の機能を実現させる。

【0054】次に、本実施の形態の具体的処理例を説明する。

【0055】始めに、受信周波数の設定を例に、本実施の形態の無線機の動作を説明する。上述の如く、アプリ

ケーション14には受信周波数を設定するための関数が用意されており（ここで関数名を”set-receive-frequency”とする）、次のように記述されている。

【0056】

#----Application Source Program

\$freq_recv = 1.9e9;

set_receiving_frequency(\$freq_recv);

#----

この例は1900MHzの信号を受信するための命令を示しており、実行時には上記関数set_receiving_frequencyは変数freq_recvを引数としてOS11に引き渡す。実際に受信周波数を設定するためにはシンセサイザの発振周波数の設定を行う必要がある。このような受信周波数→シンセサイザ発振周波数への変換はハードウェアの構成によって異なるものであるため、アプリケーションの実行時にライブラリ13が参照されそこに用意された関数によってなされる。

【0057】たとえばライブラリ13には受信周波数の設定を受けてそれをシンセサイザ発振周波数へ変換する関数（仮に”receiving-freq-to-synthesizer”とする）が引かれ、次のようになる。

【0058】

#----Library

\$freq_synth = receiving_freq_to_synthesizer (1.9e9);

#----

なお上記の例ではアルファベット文字で記述されているが実際には上記の意味を持つ機械語である。

【0059】次に図4を用いて、動作例を説明する。図4は、本実施の形態に係る無線機のハードウェア構成を信号の流れに沿って示す図である。図中1はアンプ、2はミキサ、3はシンセサイザ、4はA/D変換器であり、アンテナから受信された信号はアンプ1によって増幅された後、ミキサ2によってシンセサイザ3から供給される信号と乗算されて低いIF周波数へ周波数変換され、A/D変換器でデジタル信号へ変換される。

【0060】ここで例えばIFが380MHzである場合、シンセサイザの発振周波数は受信周波数（1.9GHz）とIF（380MHz）との差であるところの1520MHzとなる。受信周波数からシンセサイザの発振周波数への変換が、コンパイルされたアプリケーションプログラムの実行時にライブラリ13によってなされる。

【0061】このような受信周波数→シンセサイザ発振周波数への変換はハードウェアに依存した部分であるために、変換処理が直接アプリケーションプログラムに記述されることはなく、実行時にライブラリの部分で変換処理に相当する処理が行われる。

【0062】さらに、コンパイルされたバイナリがライブラリ13を参照して実際にOS11上で動作すると

き、上記ライブラリによって決定されたシンセサイザの発振周波数の設定はOS 11を介してデバイスドライバ12の処理によって実行される。

【0063】そのときにデバイスドライバ12へ渡す命令はやはりライブラリ13によって生成される。例えば次のような命令となる。

【0064】

```
#----Output to device driver through OS
$freq_pdet=2.0e5; #位相比較周波数は200,000Hz(=200
kHz)
$freq_ref=1.0e7; #基準周波数は10,000,000Hz(=10MHz)
$1 = $freq_synth; #変数$1は$freq_synthに格納された値
$2 = $freq_pd; #変数$2は位相比較周波数
$3 = $freq_ref; #変数$3は基準周波数
out_to_syn($1, $2, $3); #シンセサイザに対して変数
$1, $2, $3を出力
#----
```

ここまでの動作はコンパイルされたアプリケーションが実行時にライブラリ13を参照することによってなされる。

【0065】次に、この命令を受けたOS 11はデバイスドライバ12のうちシンセサイザ3を駆動するデバイスドライバに対してパラメータ\$1、\$2、\$3を引き渡す。out_to_synという関数がシンセサイザ3のデバイスドライバに対する命令であることを意味している。

【0066】OS 11はシンセサイザ3のデバイスドライバに出力するためのアドレスを保持しており、この命令を受けたOS 11は予め定められたシンセサイザのI/Oポートに対して3つのパラメータ(\$1、\$2、\$3)を出力する。

【0067】ここでは\$1が発振周波数、\$2が位相比較周波数(200kHz)、\$3が基準周波数(10MHz)の値が入力されている例を示している。ここでは位相比較周波数や基準周波数はハードウェア10の構成に応じて自動的に設定されるためにアプリケーション14によって明示的に与えられることはなく、ライブラリ13によって自動的に与えられる。

【0068】次にシンセサイザ3の詳細を示すハードウェア構成の例を図5に示す。図5中18はリファレンスカウンタ、19はプログラマブルカウンタ、23はスローカウンタ、24はプリスケアラであり、リファレンスカウンタ18は入力された水晶発振器等の基準信号を分周して位相比較器に供給する。

【0069】またプログラマブルカウンタ19、スローカウンタ23、プリスケアラ24はこれらを組み合わせることによって全体として比較的大きな分周をおこなう可変分周器として動作する。

【0070】これらの分周器の動作は、一般的な分周器

と同様であるので詳細説明は省略する。またリファレンスカウンタ18、プログラマブルカウンタ19、スローカウンタ23、プリスケアラ24はPLL-IC25に内蔵されている。

【0071】ここで、OS 11からパラメータを受け取ったデバイスドライバ12は、\$1、\$2、\$3のパラメータの値をシンセサイザ3で利用しているPLL-IC25の設定信号に変換して送出するという動作をおこなう。

10 【0072】例えばこの場合、

- ・発振周波数と位相比較周波数の比が1900MHz / 200kHz = 9500 → 比較周波数側の総分周数が9500

- ・基準周波数と位相比較周波数の比10MHz / 200kHz = 50 → 基準周波数側の分周数が50

と算出される。シンセサイザ3の構成が図5に示されるものであった場合、これらの値をもとに、分周器の分周数が計算され、

- ・プログラマブルカウンタ22のカウント値が148

- ・スローカウンタ23のカウント値が28

- ・リファレンスカウンタ21のカウント値が50

となるような信号がデバイスドライバ12からシンセサイザ3のPLL-IC25に対して送出され、シンセサイザ3の発振周波数が最終的に設定される。

【0073】デバイスドライバ12ではこれらの値が2値のデジタル信号に変換され、次のようなビット列が生成される。(カウンタは全て8ビットを仮定)

#----Synthesizer Data

#----Programable Counter

30 10010100 #148を二進数に変換

#----Swallow Counter

00011100 #28を二進数に変換

#----Reference Counter

00110010 #50を二進数に変換

#----

このあと、デバイスドライバ12は生成されたビット列をCPU9に送出する。CPU9は与えられたデータをハードウェアすなわちPLL-IC25に対してシリアルで、例えば図6に示すようなdataとclockとstrobe信号を送出する。

40 【0074】なお本実施の形態ではCPU9から直接シンセサイザ3へ信号を出力する例を挙げているが、I/O用の専用デバイスを設けそこから信号出力をおこなってもよい。

【0075】図6はCPU9からシンセサイザ3に対して送出されるデータ列の例を示したものであり、上記の例のプログラマブルカウンタに分周数148を設定する例を示している。

50 【0076】信号は3種類のシリアル信号として送られ、データの内容を示すDATA、データの区切りを示

すCLOCK、そしてデータの送出終了を示すSTROBEからなる。

【0077】データはCLOCK信号の立ち下がりでラッチされる。順に10010100というデータをこのクロックの立上りに同期させて送出することで所望の分周数がプログラブルカウンタに送られる。

【0078】データ送出終了後にSTROBE信号が立ち上がってデータが終了したことが示されプログラブルカウンタの分周数として設定される。

【0079】このような動作を、スワローカウンタ、リファレンスカウンタに対して同様におこなうことによって、シンセサイザには所望の分周数が設定される。これによって最終的に所望の発振周波数が得られ、所望の周波数の信号の受信が可能となる。

【0080】以上のような段階を経て、最初アプリケーションプログラムに記述された、「1900MHzの信号を受信せよ」という命令が、最終的に、ロジックデータとしてシンセサイザのPLL-ICに送出されるという命令に変換され実際に受信が可能となる。

【0081】このようにアプリケーションプログラムには物理層制御の命令のみを記述するだけで、ライブラリ/OS/デバイスドライバ/ハードウェアが適切に動作することによって所望の動作を実行することが可能となる。

【0082】以上のように無線機を構成することによって、アプリケーションプログラムから無線機側を見たインターフェイス上には、動作対象がハードウェアであるかソフトウェアであるかという区別が無いため、ハード/ソフトを意識しないプログラミングが可能となり、また、ハードウェア側から見れば、アプリケーションプログラムとのインターフェイスを統一的に扱うことによってあらゆるアプリケーションプログラムが汎用的に利用可能となる。

【0083】また、パーソナルコンピュータ等の計算機では無線の物理層制御をアプリケーションプログラムからハードウェアに依存せずに実行することは不可能であったが、本発明の無線機を用いることによってアプリケーションプログラムからハードウェアに依存しない無線物理層制御が可能となる。

【0084】なお、一般的なコンピュータシステムにおいても、アナログ回路の制御は、上述した従来例と同様に、そのアナログ回路を制御するための専用のハードウェアを設けて行うようになっており、本実施の形態のようにライブラリを用いてアナログ回路を制御する技術は考えられていない。

【0085】図7及び図8は本発明の他の実施の形態を示すフローチャートである。

【0086】図16の実施の形態における無線通信装置のハードウェアは、アナログ部とデジタル部からなるが、無線機能をアナログ部で実現するか、デジタル部

で実現するかは本ハードウェア上で動作するソフトウェアによって選択することが可能である。たとえば、アナログ部にベースバンドのアクティブフィルタが具備されており、デジタル部にFPGAによるFIRフィルタが具備されているような場合を考えると、アナログベースバンドフィルタをローパスフィルタとして用い、入力受信信号の余分な周波数成分を取り除くことが可能であるが、このアナログフィルタをスルーで用い、デジタル部のFIRフィルタに適切なタップ係数を付与することによって、全く同一又は類似した周波数特性を持つフィルタを形成可能である。

【0087】もちろん同一機能を実現できる範囲（たとえばローパスフィルタの例ではカットオフ周波数や次数等）はハードウェアによって制限されるが、特性的にオーバーラップする部分がある。本発明におけるアプリケーションプログラムはハードウェアの区別なく記述されているので、ある機能を実現する回路部分（以下、機能ブロックという）がアナログ部とデジタル部のいずれにあっても実現可能である場合には、その割り振りが問題となる。機能ブロック、即ち、ある機能を実現するハードウェアの選択をライブラリ13がおこなうことが本実施の形態の特徴である。

【0088】まず第1の方法は、ライブラリはアプリケーションプログラムの命令をできる限りアナログブロックで実現する方向で全体をコンフィギュレーションする方法である。たとえば機能Aがアナログ部でもデジタル部でも実現できる場合に、その機能を優先的にアナログ部に割り当ててゆくという方法である。

【0089】図7はこの方法の詳細な手順を示している。まずアプリケーションプログラムのコンパイル時に、アプリケーションプログラム内に記述された無線機能に関する命令を解釈する（ステップS1）。次にその機能がデジタルとアナログ部のいずれかまたは両方で実現可能であるかどうかを調べる（ステップS2）。このときもしデジタル部あるいはアナログ部の一方でしか実現できない場合には実現できるブロックで実行するようにコンパイルされるが、デジタル部とアナログ部の両方で実現可能である場合、優先的にその機能をアナログ部で実現するようにコンパイルされる（ステップS4）。またもしデジタル部でもアナログ部でも実現できない場合は異常終了する。

【0090】このようにコンパイルすることによって、たとえばデジタル部で実現するとデジタル回路の消費電力が必要であるのに対して、アナログ部で実現した場合、たとえばその機能がパッシブ回路によって実現可能であるような場合には消費電力を0にすることが可能となり、無線装置の低消費電力化が可能となる。

【0091】次に第2の方法は、第1の方法とは逆に、ライブラリはアプリケーションプログラムの命令をできる限りデジタルブロックで実現する方向で全体をコン

フィギュレーションする方法である。

【0092】図8はこの方法の詳細な手順を示している。まずアプリケーションプログラムのコンパイル時に、アプリケーションプログラム内に記述された無線機能に関する命令を解説する（ステップS1）。次にその機能がデジタルとアナログ部のいずれかまたは両方で実現可能であるかどうかを調べる（ステップS2）。このときもしデジタル部あるいはアナログ部の一方でしか実現できない場合には実現できるブロックで実行するようにコンパイルされる。デジタル部とアナログ部の両方で実現可能である場合、優先的にその機能をデジタル部で実現するようにコンパイルされる（ステップS5）。またもしデジタル部でもアナログ部でも実現できない場合は異常終了する。

【0093】このようにコンパイルすることによって、たとえばアナログ部で実現すると特性があまりよくない場合に、デジタル部で実現した場合、理想的な特性が実現でき、無線装置の全体の良好な特性が得られる。

【0094】上記のデジタル部／アナログ部の実現手段の選択は、すべての無線機能に関して一律に設定されるのではなく、無線機能ブロックごとに区分けし、たとえばフィルタはデジタル部優先、周波数変換器はアナログ部優先などとあらかじめ決めておくことによって、全体として特性の向上を実現することが可能となり、さらにこの選択に関してはアプリケーションプログラムはまったく関与しないため、アプリケーションの開発は独自に可能で、ハードウェアに依存した無線機能実現の手段はコンパイラやライブラリが自動的に割り当てられるため理想的な構成が実現できる。

【0095】以上述べた、アナログ部／デジタル部の機能の選択は、その法則（デジタルから割り当てるか、アナログから割り当てるか）をあらかじめライブラリに記述することによって、アプリケーションプログラムの内容によって、自動的に振り分けることが可能となる。たとえば上述のようにアナログのローパスフィルタに比べて、デジタルフィルタの消費電力が大きい場合、可能な限りアナログフィルタで処理した方が、無線機全体の低消費電力化が図れる。このような場合はライブラリによって、優先的にアナログ部に処理されるように、無線機が制御される。

【0096】この場合、ライブラリは、アナログ部とデジタル部の両方で実現可能な無線機能についてのテーブルを保持し、テーブルに記述された無線機能の限界性能、および、法則を参照することによって、ある機能をアナログ部又はデジタル部のいずれに割り当てるかを決定する。

【0097】上記実施の形態においては、機能ブロック別にアナログ／デジタル部での実現の振り分け方法を決めたが、信号処理速度によって別けてももちろんよい。

【0098】さらに本実施の形態のバリエーションとして次のような実施の形態が挙げられる。すなわちアナログ部、デジタル部両方で機能が実現できる場合、ある判定条件を元にどちらで機能を実現できるか決定する方法である。図9はこの方法を示すフローチャートである。

【0099】まずアプリケーションプログラムのコンパイル時に、アプリケーションプログラム内に記述された無線機能に関する命令を解説する（ステップS1）。次にその機能がデジタルとアナログ部のいずれかまたは両方で実現可能であるかどうかを調べる（ステップS2）。このときもしデジタル部あるいはアナログ部の一方でしか実現できない場合には実現できるブロックで実行するようにコンパイルされる。デジタル部とアナログ部の両方で実現可能である場合、次のステップS3へ進む。次のステップS3では、あらかじめ定められた判定基準をもとにデジタル部で実行するかアナログ部で実行するかを判断し、ステップS4又はステップS5において、アナログ部又はデジタル部のいずれかで機能を実現する。

【0100】上記判断基準としては次のようなものが挙げられる。

- 【0101】（１）消費電力が少ない方を選択する
- （２）特性がよい方を選択する
- （３）遅延時間が少ないほうを選択する
- （４）デジタル部とアナログ部間の信号の往復が少ない方を選択する

上記（１）はアナログ部、デジタル部で実現した場合の消費電力を見積もりそれをもとに少ないほうを選択する。（２）はたとえば所望の特性はアナログ部、デジタル部いずれでも選択可能であるが、デジタル部の方がより良好な特性を実現可能である場合はデジタル部を選択する。（３）は機能を実現するための信号処理時間を見積もり、より遅延時間の少ないほうを選択する。

（４）はたとえば所望の無線信号処理機能をデジタル部で実現した場合に、受信信号がいったんアナログ部を出てデジタル部へ入り所望の処理を行い、再度アナログ部へ入力する必要があるような場合は、信号のデジタル部とアナログ部間の往復を避けるためにアナログ部で処理するように選択する。

【0102】以上のように動作することによって、最適な無線部の構成が実現可能となる。

【0103】図10は本発明の第3の実施の形態において採用されるアナログの機能ブロック（アナログブロック）を示す説明図である。本実施の形態は、アナログブロックの構成が異なるのみであり、デジタルの機能ブロック及びソフトウェアの構成は図1と同様であり、図示を省略する。

【0104】図10において、アナログブロック131は仕様記述部132を有している。なお、図1における

全てのアナログブロックがアナログブロック 131 と同様に仕様記述部 132 を備えていてもよく、一部のアナログブロックのみが仕様記述部 132 を備えていてもよい。なお、上述したように、アナログブロックとしては、例えばアンプ、ミキサ、シンセサイザなどがある。

【0105】仕様記述部 132 は、アナログブロック 131 の仕様が記述されている。仕様記述部 132 は仕様を記憶することができれば、不揮発性のメモリ等によって構成してもよく、また、スイッチ等によって構成してもよい。仕様記述部 132 に記述されているアナログブロック 131 の仕様は、外部からのアクセスによって読み出すことができるようになっていて、例えば、図 1 では、主として CPU9 が各アナログブロックの特性を読み出すことができるようになっていて、

【0106】図 11 はアナログブロックが例えば図 1 のシンセサイザ 3 に相当するシンセサイザ 141 である場合を示す。シンセサイザ 141 に関する仕様は仕様記述部 142 に記述されている。例えば、仕様記述部 142 には、発振可能周波数、位相雑音、周波数切換時間、周波数安定度等が次のような形で書込まれている。

【0107】

発振可能周波数	1895.15~1917.95MHz
周波数切換時間	1msec
位相雑音	-27dB
周波数安定度	±2.5ppm

なお、仕様記述部 142 の仕様は、予め定義された基準に従って記述される。例えば、周波数安定度については温度範囲等の基準が、また、位相雑音についてはその積分範囲等の基準が予め設定され、これらの基準ののって仕様が記述されている。

【0108】また、図 12 にはアナログブロックが図 1 のアンプ 1 に相当するアンプ 151 である場合を示す。アンプ 151 に関する仕様は仕様記述部 152 において、例えば使用可能周波数、利得可変範囲、NF 等が次のような形で書込まれている。

【0109】

使用可能周波数	1.5~2GHz
利得可変範囲	0~40dB
NF	2.0dB

ライブラリ側は、デバイスドライバや OS 等の基本ソフトウェアを介してこれらの情報を読みとることが可能であり、必要な性能が得られるか否か等の判断材料として利用することが可能となる。例えばこれらの情報によって、ある特定のハードウェア上で、無線機全体として必要な特性が得られるかどうかの判断がライブラリ側からソフトウェア的に判断可能となる。

【0110】また、アナログ部はハードウェア的にシンセサイザ、アンプ、ミキサ等が分離しておらず、1チップ化されている場合があるが、このような場合には IC の内部に、1チップ化されているいくつかの機能ブロッ

クに関する上記の情報をまとめて記述する仕様記述部を設けるか、あるいは分散させて記述する複数の仕様記述部を設け、これらの仕様記述部を CPU 側から読み取れるように構成しておけばよい。

【0111】たとえば図 13 にはシンセサイザとアンプが 1チップ化された IC の例を示している。アナログブロック 161 は、図 1 のアンプ 1 及びシンセサイザ 3 に相当するものであり、仕様記述部 162 にはアナログブロック 161 のアンプ部とシンセサイザ部の各仕様が記述されており、CPU 側からこれらの情報をそれぞれ別々に読み出すことが可能である。

【0112】次にこれらの情報の記述方法について述べる。例えば、あらかじめアナログブロックに ROM などのメモリを作りこんでおき、そのなかに情報を書き込んでおく方法がある。この場合には、ROM の情報を引き出すための専用のデータ端子とアドレス端子とを設け、CPU からアドレスを指定することによってデータ端子からアナログブロックの特性をデジタルデータの形で出力させることができる。

【0113】図 14 は本発明の第 4 の実施の形態において採用されるアナログの機能ブロック（アナログブロック）を示す説明図である。本実施の形態は、図 10 の第 3 実施の形態に対して仕様記述部の構成が異なる。

【0114】すなわち、本実施の形態では仕様記述部に仕様そのものを記述するのではなく、そのアナログブロックの識別番号を記述しておくのである。そしてその識別番号とそのアナログブロックの仕様に関してはライブラリの中に対照表をあらかじめ用意することによって、CPU はアナログブロックの特性に関する情報を得ることができる。

【0115】図 14 において、アナログブロック 131 に確保された仕様記述部 132 は、アナログブロック 131 の識別番号が記述されている。CPU9 はアナログブロック 131 の仕様記述部 132 から識別番号を読み取り、その識別番号をもとに、ライブラリ 13 に記述された識別番号と仕様との対照表を参照することによって、必要な情報をライブラリから得ることができる。

【0116】このように構成することによって、図 10 に示した構成に比べて、情報を得るための手順は煩雑になるが、アナログブロックに記述しておく情報量を大幅に削減することができ、より好ましい方法であるといえる。仕様記述部に仕様を直接記述した場合、その情報量はたとえば仕様項目が 5、各仕様の内容の記述に 20 バイト必要であるものとする、合計 100 バイトとなるのに対し、識別番号だけであればたとえば 4 バイト程度と、25 分の 1 のサイズに抑えることが可能となる。識別番号として 4 バイトを記憶させる場合でも、32 ビット分（2 の 32 乗）の種類の識別が可能であるため十分であるといえる。

【0117】識別番号の付し方については図 15 に示す

21

ように次のような方法が考えられる。すなわち、最初のブロックに機能181（たとえばシンセサイザ、ミキサ、アンプなど）、次のブロックに製造業社名182、そして最後に識別番号183とする。機能はたとえば4ビットのバイナリで下記のように表現する。

【0118】

0000	ミキサ
0001	受信用アンプ
0010	シンセサイザ
0011	送信用アンプ
0100	アッテネータ
.....

また、次の4ビットは製造者を表す。

【0119】

0000	製造者A
0001	製造者B
0010	製造者C
0011	製造者D
0100	製造者E
.....

なお上記は一例であつたたとえば機能を表現するビット数は必ずしも4ビットである必要はなく、また製造者をあらわす部分に関しても同様であり、必要に応じたビット数をあらかじめ勘案して決定すればよい。

【0120】また本実施の形態についてはアナログ部の場合をのべたが、専用機能を有し、その特性を外部からの設定によって変更することが可能であるデジタル部、たとえばタップ係数をパラメータとして与えるタイプのデジタルフィルタや、局部発振器の発振周波数をパラメータとして与えるデジタル直交復調器などのICにも同様の仕様記述部を設けておくことは有効であることは言うまでもない。たとえばデジタルフィルタの場合は次のような仕様を記述しておくとい。

【0121】

最大タップ数	256
タップ係数の精度	16ビット
最大処理速度	50Mサンプル/秒

デジタル部は内部がデジタル回路によって構成されているため、デジタル的にその仕様を外部へ出力する構成をなすことは上述のアナログブロックの場合に比して特に製造しやすいので本実施の形態の方法によりなじんだ方法であるともいえる。

【0122】本発明においては、発明の精神及び範囲から逸脱することなく、広い範囲において異なる実施態様を、本発明に基づいて構成することができることは明白である。本発明は、添付のクレームによって限定される以外には、その特定の実施態様によって制約されない。

【0123】

22

【発明の効果】以上説明したように本発明によれば、汎用性を有し、ハードウェアアーキテクチャに依存しないアプリケーションプログラムを利用可能となるという効果を有する。

【図面の簡単な説明】

【図1】本発明の第1の実施の形態を説明するための図である。

【図2】本発明のコンパイラの動作を示す図である。

【図3】本発明のコンパイラの動作を示す図である。

10 【図4】アナログ部の構成を示す図である。

【図5】シンセサイザの構成を示す図である。

【図6】シンセサイザ設定データを示す図である。

【図7】本発明の他の実施の形態を示すフローチャートである。

【図8】本発明の他の実施の形態を示すフローチャートである。

【図9】本発明の他の実施の形態を説明するためのフローチャートである。

20 【図10】本発明の第3の実施の形態において採用されるアナログの機能ブロック（アナログブロック）を示す説明図である。

【図11】アナログブロックが例えば図1のシンセサイザ3に相当するシンセサイザ141である場合を示す説明図である。

【図12】アナログブロックが図1のアンプ1に相当するアンプ151である場合を示す説明図である。

【図13】シンセサイザとアンプが1チップ化されたICの例を示す説明図である。

30 【図14】本発明の第4の実施の形態において採用されるアナログの機能ブロック（アナログブロック）を示す説明図である。

【図15】識別番号の付し方を説明するための説明図である。

【図16】従来例を説明するための図である。

【符号の説明】

1・・・アンプ

2・・・ミキサ

3・・・シンセサイザ

4・・・AD変換器

40 5・・・DA変換器

7・・・DSP

9・・・CPU

10・・・ハードウェア

11・・・OS

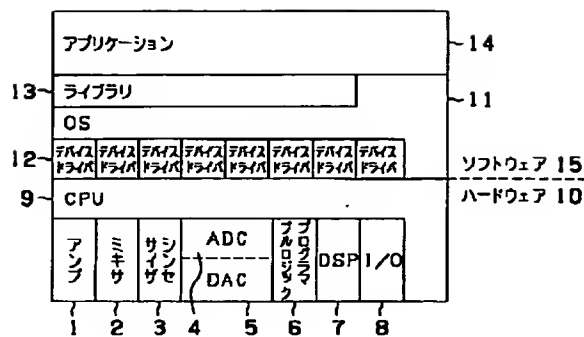
12・・・デバイスドライバ

13・・・ライブラリ

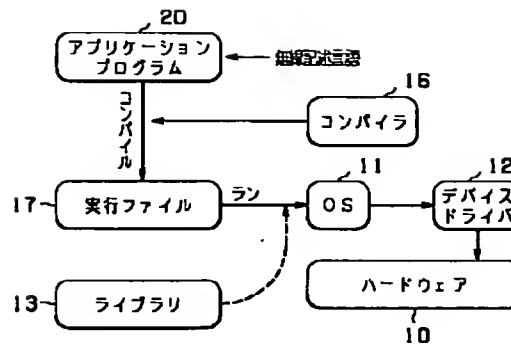
14・・・アプリケーションプログラム

15・・・ソフトウェア

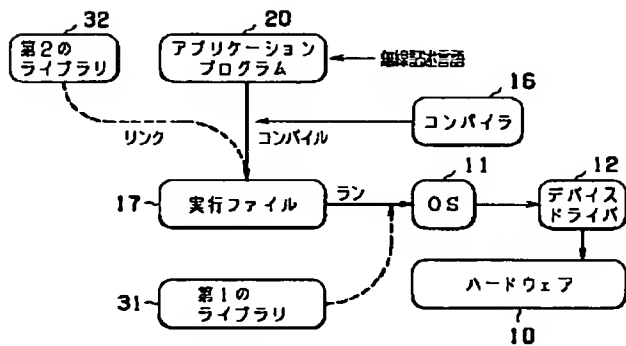
【図 1】



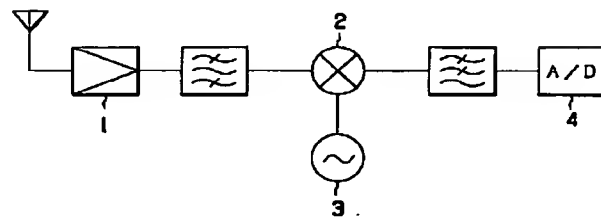
【図 2】



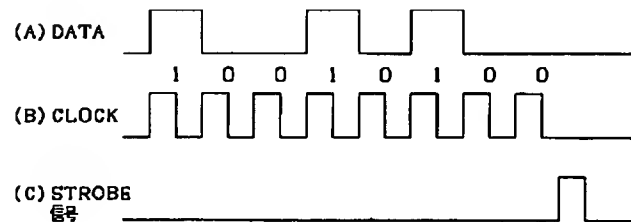
【図 3】



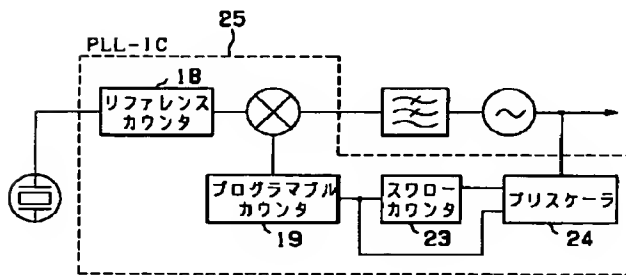
【図 4】



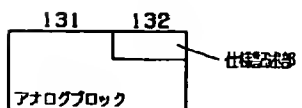
【図 6】



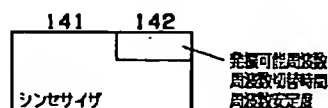
【図 5】



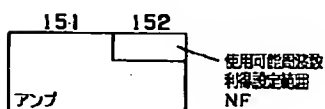
【図 10】



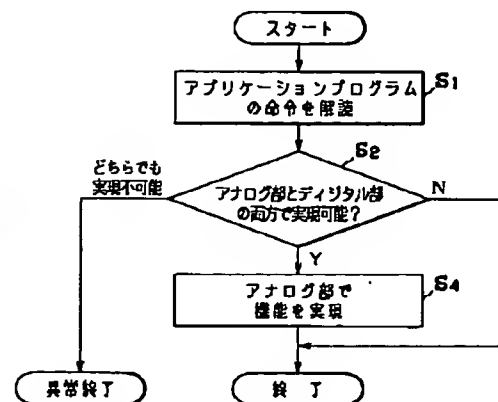
【図 11】



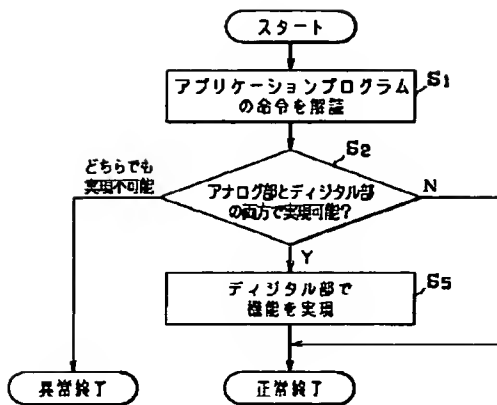
【図 12】



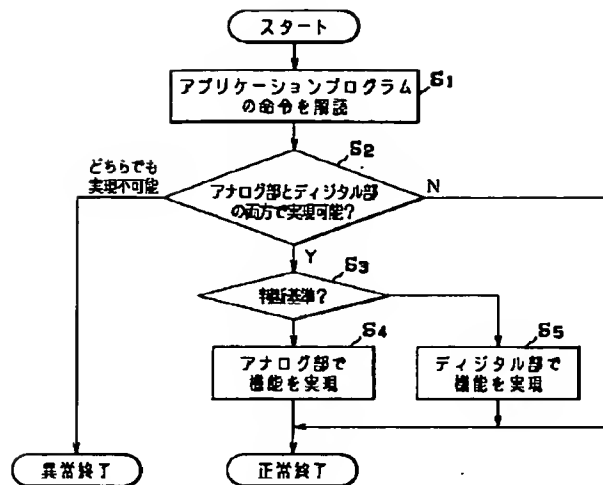
【図 7】



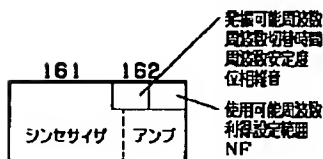
【図 8】



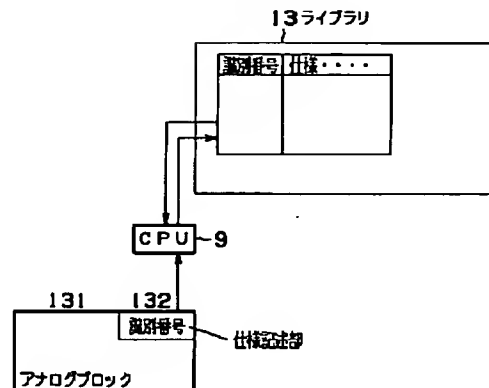
【図 9】



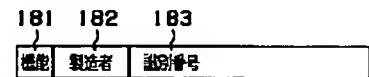
【図 13】



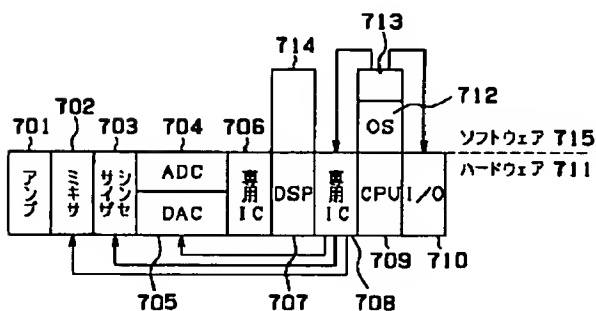
【図 14】



【図 15】



【図 16】



フロントページの続き

(72)発明者 富澤 武司
神奈川県川崎市幸区小向東芝町 1 番地 株
式会社東芝研究開発センター内

(72)発明者 行方 稔
神奈川県川崎市幸区小向東芝町 1 番地 株
式会社東芝研究開発センター内

(72)発明者 鶴見 博史
神奈川県川崎市幸区小向東芝町1番地 株
式会社東芝研究開発センター内

(72)発明者 玉田 雄三
神奈川県川崎市幸区小向東芝町1番地 株
式会社東芝研究開発センター内